

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ

Заведующий кафедрой

_____ Г. М. Цибульский

подпись

«___» _____ 2017 г.

БАКАЛАВРСКАЯ РАБОТА

09. 03. 02 «Информационные системы и технологии»

База знаний системы поддержки принятия решений
в сельскохозяйственном производстве

Руководитель _____ ст. преподаватель каф. СИИ К. В. Раевич
подпись, дата

Выпускник _____ Р. Е. Гульцев
подпись, дата

Консультант _____ проф. канд. техн. наук Ю. А. Маглинец
подпись, дата

Нормоконтролер _____ М. А. Аникьева
подпись, дата

Красноярск 2017

Продолжение титульного листа бакалаврской работы по теме
«База знаний системы поддержки принятия решений в сельскохозяйственном
производстве»

Нормоконтролер _____
подпись, дата

М. А. Аникьева

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ

Заведующий кафедрой

_____ Г. М. Цибульский

подпись

«___» _____ 2017 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврской работы

Студенту Гульцеву Роману Евгеньевичу

Группа КИ13-14Б, направление 09.03.02 «Информационные системы и технологии», профиль 09.03.02.04 «Информационные системы и технологии в медиаиндустрии».

Тема выпускной квалификационной работы «База знаний системы поддержки принятия решений в сельскохозяйственном производстве».

Утверждена приказом по университету № 2567/с от 01. 03. 2017.

Руководитель ВКР Раевич К. В. старший преподаватель кафедры систем искусственного интеллекта ИКИТ СФУ.

Исходные данные для ВКР: методические указания руководителя.

Перечень разделов ВКР: введение; глава 1. Теоретическая часть; выводы по главе 1; глава 2. Программная часть; выводы по главе 2; заключение; список использованных источников; приложение (плакаты презентации); приложение (диплом участия в конференции).

Руководитель ВКР

К. В. Раевич

подпись

Задание принял к исполнению

Р. Е. Гульцев

подпись

«29» февраля 2017 г.

График

выполнения выпускной квалификационной работы студентом направления 09. 03. 02 «Информационные системы и технологии», профиля 09. 03. 02. 04 «Информационные системы и технологии в медиаиндустрии».

График выполнения выпускной квалификационной работы приведен в таблице 1.

Таблица 1 — График выполнения этапов ВКР

Наименование этапа	Срок выполнения этапа	Результат выполнения этапов	Примечание руководителя (отметка о выполнении этапа)
Ознакомление с целью и задачами работы	07. 03. 17 - 13. 03. 17	Краткое эссе по теме ВКР	Выполнено
Сбор литературных источников	14. 03 - 20.03	Список источников литературы	Выполнено
Анализ собранных литературных источников	21. 03. 17 - 27.03. 17	Реферат о проблемно-предметной области	Выполнено
Уточнение и обоснование актуальности цели и задач ВКР	28. 03. 17 - 3. 04. 17	Окончательная формулировка цели и задач ВКР	Выполнено
Решение задач ВКР	04. 04. 17 - 25. 05. 17	Доклад с презентацией по теме ВКР	Выполнено
Компоновка отчета по результатам решения задач ВКР	26. 05. 17 - 2. 06. 17	Отчет по результатам решения задач ВКР	Выполнено
Предварительная защита результатов	7. 06. 17	Доклад и презентация по проделанной работе	Выполнено

Окончание таблицы 1— График выполнения этапов ВКР

Наименование этапа	Срок выполнения этапа	Результат выполнения этапов	Примечание руководителя (отметка о выполнении этапа)
Нормоконтроль	2. 06. 17 - 20. 06. 17	Пояснительная записка	
Защита ВКР	22. 06.17 - 23. 06. 17	Доклад и презентация по результатам бакалаврской работы	

Руководитель ВКР

К. В. Раевич

подпись

Студент гр. КИ13-14Б

Р. Е. Гульцев

подпись

СОДЕРЖАНИЕ

Введение.....	7
Глава 1 Обзор предметной области.....	9
1.1 Обзор систем поддержки принятия решений.....	9
1.2 Методы приобретения знаний.....	10
1.3 Анализ знаний.....	11
1.4 База знаний.....	13
1.5 Аналоги СППР использующие БЗ.....	17
1.6 Дескрипционные логики.....	23
1.7 Рассуждение по прецедентам.....	24
Выводы по главе 1.....	27
Глава 2. Использование программных средств	28
2.1 Создание прецедентов.....	28
2.2 Связь свойств прецедентов.....	31
2.3 Создание системы решения по прецедентам.....	32
2.4 Результат работы программы.....	36
Выводы по главе 2.....	38
Заключение	40
Список сокращений	41
Список использованных источников	42
Приложение А Слайды презентации.....	44
Приложение Б Сертификат участника.....	50

ВВЕДЕНИЕ

В процессе принятия инвестиционных и управленческих решений в сфере управления агропромышленным комплексом (далее АПК) существенную роль играет обеспечение лица, принимающего решения (далее ЛПР), релевантной актуальной информацией. На современном уровне развития информационных технологий такая информация может быть предоставлена при помощи систем поддержки принятия решений (далее СППР).

Научный подход к обоснованию принимаемых решений является гарантией успеха в управлении сельскохозяйственным производством за счет глубокого анализа поставленных проблем на основе объективных данных с использованием различных научных методов принятия решений. Однако решение сложных, комплексных проблем в условиях неопределенности и неполноты информации в значительной мере осложняет принятие высокоэффективных решений, направленных на повышение конкурентоспособности и улучшение качества производства [1].

В процессе выполнения работы выбрано решение на основе прецедентов. Основной целью использования аппарата прецедентов в СППР является выдача готового решения ЛПР для текущей ситуации на основе прецедентов, которые уже имели место в прошлом при управлении данным объектом или процессом. Поиск решения на основе прецедентов заключается в определении степени сходства текущей ситуации с ситуациями прецедентов из базы знаний (далее БЗ). При этом учитываются веса параметров для ситуации из БЗ, заданные экспертом. Степень сходства зависит от близости текущей ситуации к ситуации прецедента. Рассуждение на основе прецедентов может не привести необходимому решению проблемной ситуации в случае отсутствия подобной ситуации в БЗ.

Таким образом, пользователю СППР предоставлен широкий набор возможностей для выработки оптимальных решений для управления сельскохозяйственным предприятием. Синтезирував адаптивную (применимую в конкретном хозяйстве) агротехнологию, пользователь имеет возможность оценить ее по экономическим и экологическим критериям и в зависимости от стратегии развития предприятия. Важным является то, что агротехнические операции, включенные в агротехнологию, можно проводить в режиме точного земледелия (дифференцированно) за счет реализованных механизмов генерации карт-заданий и таблиц агротребований. По мере наполнения базы знаний, система будет полезной для любого сельскохозяйственного предприятия и может стать незаменимым консультантом для агрономов и руководителей.

Целью выпускной квалификационной работы является создание БЗ СППР в сельскохозяйственном производстве.

Задачи, поставленные для достижения цели, на выпускную квалификационную работу, были следующими:

1. обзор предметной области;
2. создание прецедентов для БЗ;
3. реализация прецедентов БЗ в программном средстве JColibri.

Глава 1 Обзор предметной области

1.1 Обзор систем поддержки принятия решений

Система поддержки принятия решений — компьютерная автоматизированная система, целью которой является помощь людям, принимающим решение в сложных условиях для полного и объективного анализа предметной деятельности [2]. Типичные применения СППР включают в себя такие задачи как медицинская диагностика, локализация неисправностей в оборудовании и интерпретации результатов изменений. Системы поддержки принятия решений должны решать задачи, требующие для своего решения экспертных знаний в некоторой конкретной области. Поэтому эти системы так же называют системами, основанными на знаниях. Система должна также уметь каким-то образом объяснять свое поведение и свои решения пользователю, так же, как это делает эксперт-человек. Это особенно необходимо в областях, для которых характерна неопределенность, неточность информации (например, в медицинской диагностике). В этих случаях способность к объяснению нужна для того, чтобы повысить степень доверия пользователя к советам системы, а также для того, чтобы дать возможность пользователю обнаружить возможный дефект в рассуждениях системы. В связи с этим в СППР следует предусматривать дружественное взаимодействие с пользователем, которое делает для пользователя процесс рассуждения системы «прозрачным».

В самом общем случае для того, чтобы построить экспертную систему, необходимо разработать механизмы выполнения следующих функций системы:

1. решение задач с использованием знаний о конкретной предметной области возможно, при этом возникнет необходимости иметь дело с неопределенностью;

2. взаимодействие с пользователем, включая объяснение намерений и решений системы во время и после окончания процесса решения задачи.

БЗ содержит знания, относящиеся к конкретной прикладной области, в том числе отдельные факты, правила, описывающие отношения или явления, а также, возможно, методы, эвристики и различные идеи, относящиеся к решению задач в этой прикладной области. Взвешенные достоверные оценки базируются на знании экспертов в областях: экологии, сельском хозяйстве, экономике. Построение системы оценок специфично для конкретной решаемой задачи и для конкретной территории, что затрудняет непосредственное привлечение экспертов. Для отчуждения экспертных знаний от их носителей и представления их в формализованном виде, доступном для использования, применяются методы «инженерии знаний».

Поэтому база знаний является ключевым блоком системы. Знания хранятся в системе в форме онтологии, которая содержится в базе знаний и поддерживает средства обеспечения целостности информации, в том числе средства управления версиями и изменениями.

1.2 Методы приобретения знаний

Приобретение знаний реализуется с помощью двух функций: получения информации извне и ее систематизации. При этом в зависимости от способности системы обучения к логическим выводам возможны различные формы приобретения знаний, а также различные формы получаемой информации.

Форма представления знаний для их использования определяется внутри системы, поэтому форма информации, которую она может принимать, зависит от того, какие способности имеет система для формализации информации до уровня знаний.

Если обучающаяся система совсем лишена такой способности, то человек должен заранее подготовить все, вплоть до формализации информации, т. е. чем выше способности машины к логическим выводам, тем меньше нагрузка на человека.

Функции, необходимые обучающейся системе для приобретения знаний, различаются в зависимости от конфигурации системы. Если база знаний пополняется знаниями о стандартной форме их представления, то этими знаниями также можно воспользоваться. Следовательно, от функций обучения требуется преобразование полученной извне информации в знания и пополнение ими базы знаний.

1.3 Анализ знаний

Анализ знаний опирается на поиск необходимых данных в БЗ. Основная задача любого поиска состоит в том, чтобы определить маршрут, по которому вы будете перемещаться с настоящей позиции к вашей цели. Если начать поиск с текущей позиции и продолжать его, пока не наткнемся на желаемый результат, - это так называемый прямой поиск или поиск снизу вверх. Если мысленно поставить себя в то место, где необходимо очутиться в результате поиска и определить маршрут, двигаясь в обратном направлении, т.е. туда, где действительно находимся в настоящий момент, - это поиск в обратном направлении или поиск сверху вниз. Обратим внимание на то, что, определив маршрут в результате обратного поиска, все же предстоит добраться до своей цели. Несмотря на то, что сейчас движение происходит вперед, это не является прямым поиском, т.к. поиск уже был осуществлен ранее, причем в обратном направлении. Эти же противопоставления можно рассмотреть на примере систем с встроенными правилами.

Представим себе, что правило состоит из набора антецедентов и набора следствий. Когда система определяет, что все антецеденты определенного правила удовлетворены, это правило вызывается и выполняется, выполняется ли каждое вызванное правило зависит от специфики конкретной системы. После этого в базу знаний заносятся утверждения, полученные в результате выполнения правила, и выполняются соответствующие операции.

Данный процесс происходит вышеописанным образом, независимо от того, применяет ли система прямой или обратный логический анализ. Чтобы проиллюстрировать различия между ними, следует отдельно рассмотреть процедуру активации правила. Вызываются только активированные правила. При прямом логическом анализе (снизу вверх), когда в систему добавляются новые данные, они сравниваются со всеми антецедентами всех правил. Если данные соответствуют антецеденту правила, то это правило активируется (если оно еще не является активированным), и если подобраны все антецеденты определенного правила, то оно вызывается. Утверждения, полученные в результате выполнения правила, заносятся в базу знаний и рассматриваются в качестве новых данных, сравниваются с антецедентами и могут вызвать активацию и вызов дополнительных правил. При обратном логическом анализе (сверху вниз) при добавлении данных правила не активируются. Когда система получает запрос, он сравнивается со всеми следствиями всех правил. Если запрос совпадает со следствием, то это правило активируется, а все его антецеденты рассматриваются в качестве вторичных запросов и могут вызвать активацию дополнительных правил. Когда запрос соответствует не ограниченному условием утверждению базы знаний, на него поступает ответ, и если этот запрос исходил от антецедента, считается, что он удовлетворяет последнему. Когда все антецеденты некоторого правила будут удовлетворены, правило вызывается и выполняется.

При выполнении правила осуществляется ответ на запросы и теперь другие антецеденты считаются удовлетворенными и могут вызываться соответствующие им правила.

Выбор вида анализа (сверху вниз или снизу вверх) зависит от конфигурации дерева, по которому осуществляется поиск. Если в среднем каждому элементу следует большее количество элементов, нежели предшествует, то анализ сверху вниз (или обратный анализ) будет более эффективным и наоборот.

1.4 База знаний

База знаний наиболее важная компонента системы поддержки принятия решений, на которой основаны ее «интеллектуальные способности». В отличие от всех остальных компонент СППР, база знаний - «переменная» часть системы, которая может пополняться и модифицироваться инженерами знаний и опыта использования СППР, между консультациями, а в некоторых системах и в процессе консультации. Существует несколько способов представления знаний в СППР, однако общим для всех них является то, что знания представлены в символьной форме, элементарными компонентами представления знаний являются тексты, списки и другие символьные структуры. Тем самым, в СППР реализуется принцип символьной природы рассуждений, который заключается в том, что процесс рассуждения представляется как последовательность символьных преобразований.

Наиболее распространенный способ представления знаний - в виде конкретных фактов и правил, по которым из имеющихся фактов могут быть выведены новые [3].

Факты представлены, например, в виде троек:

1. АТРИБУТ ;
2. ОБЪЕКТ ;
3. ЗНАЧЕНИЕ.

Такой факт означает, что заданный объект имеет заданный атрибут (свойства) с заданным значением. Например, тройка:

1. ПОЛЕ;
2. РАССТОЯНИЕ;
3. 37;

представляет собой факт «земля сельскохозяйственного назначения, обозначаемая ПОЛЕ, находится на РАССТОЯНИИ равном 37». В более простых случаях факт выражается неконкретным значением атрибута, а каким либо простым утверждением, которое может быть истинным или ложным.

Правила в базе знаний имеют вид:

ЕСЛИ A ТО S,

где A - условие;

S - действие.

Действие S выполняется, если A истинно. Наиболее часто действие S, так же, как и условие, представляет собой утверждение, которое может быть выведено системой (то есть становится ей известной), если истинно условие правила A.

Правила в базе знаний служат для представления эвристических знаний (эвристик), т. е. неформальных правил рассуждения, вырабатываемых экспертом на основе опыта его деятельности.

Простой пример правила из повседневной жизни:

ЕСЛИ поле рядом, ТО быстрее соберем урожай.

В качестве условия A может выступать либо факт (как в данном примере), либо несколько фактов A_1, \dots, A_N , соединенные логической операцией и:

A_1 и A_2 и ... и A_N .

В математической логике такое выражение называется конъюнкцией. Оно считается истинным в том случае, если истинны все его компоненты. Пример предыдущего правила с более сложным условием:

ЕСЛИ поле рядом и в подчинении много техники, ТО быстрее соберем урожай.

Действия, входящие в состав правил, могут содержать новые факты. При применении таких правил эти факты становятся известны системе, т.е. включаются в множество фактов, которое называется рабочим множеством. Например, если факты «Поле рядом» и «В подчинении много техники» уже имеются в рабочем множестве, то после применения приведенного выше правила в него также включается факт «Быстрее соберем урожай».

Если система не может вывести некоторый факт, истинность или ложность которого требуется установить, то система спрашивает о нем пользователя.

Например:

ВЕРНО ЛИ, ЧТО поле рядом?

При получении положительного ответа от пользователя факт «Поле рядом» включается в рабочее множество.

Существуют динамические и статические базы знаний. Динамическая база знаний изменяется со временем. Ее содержимое зависит от состояния окружающей.

В нашей системе будет использован статический подход к БЗ. Эксперт будет наполнять БЗ необходимой информацией.

Новые факты, добавляемые в базу знаний, являются результатом вывода, который состоит в применении правил к имеющимся фактам.

В системах с монотонным выводом факты, хранимые в базе знаний, статичны, то есть не изменяются в процессе решения задачи. В системах с немонотонным выводом допускается изменение или удаление фактов из базы знаний. В качестве примера системы с немонотонным выводом можно привести СППР, предназначенную для составления перспективного плана капиталовложения компании. В такой системе по вашему желанию могут быть изменены даже те данные, которые после вывода уже вызвали срабатывание каких-либо правил. Иными словами имеется возможность модифицировать значения атрибутов в составе фактов, находящихся в рабочей памяти. Изменение фактов в свою очередь приводит к необходимости удаления из базы знаний заключений, полученных с помощью упомянутых правил. Тем самым вывод выполняется повторно для того, чтобы пересмотреть те решения, которые были получены на основе подвергшихся изменению фактов.

Проанализировав информацию о БЗ можно утверждать, что описание проблемы должно содержать всю информацию, необходимую для достижения цели вывода то есть выбора наиболее подходящего решения. Проблема представления знаний это, прежде всего проблема выбора информации, которую надо включать в описание, нахождение соответствующей структуры для описания содержания знаний, а также определения, каким образом должна быть организована и индексирована БЗ для эффективного поиска и многократного использования.

Для наиболее детального рассмотрения БЗ рассмотрим СППР содержащие в себе знания, для того, чтобы выбрать наиболее подходящую модель реализации поставленной задачи.

1.5 Аналоги СППР использующие БЗ

Функциональная структура СППР старшего преподавателя кафедры системы искусственного интеллекта К. В. Раевич и доктора технических наук И. В. Зенькова [2] представлена на рисунке 1:

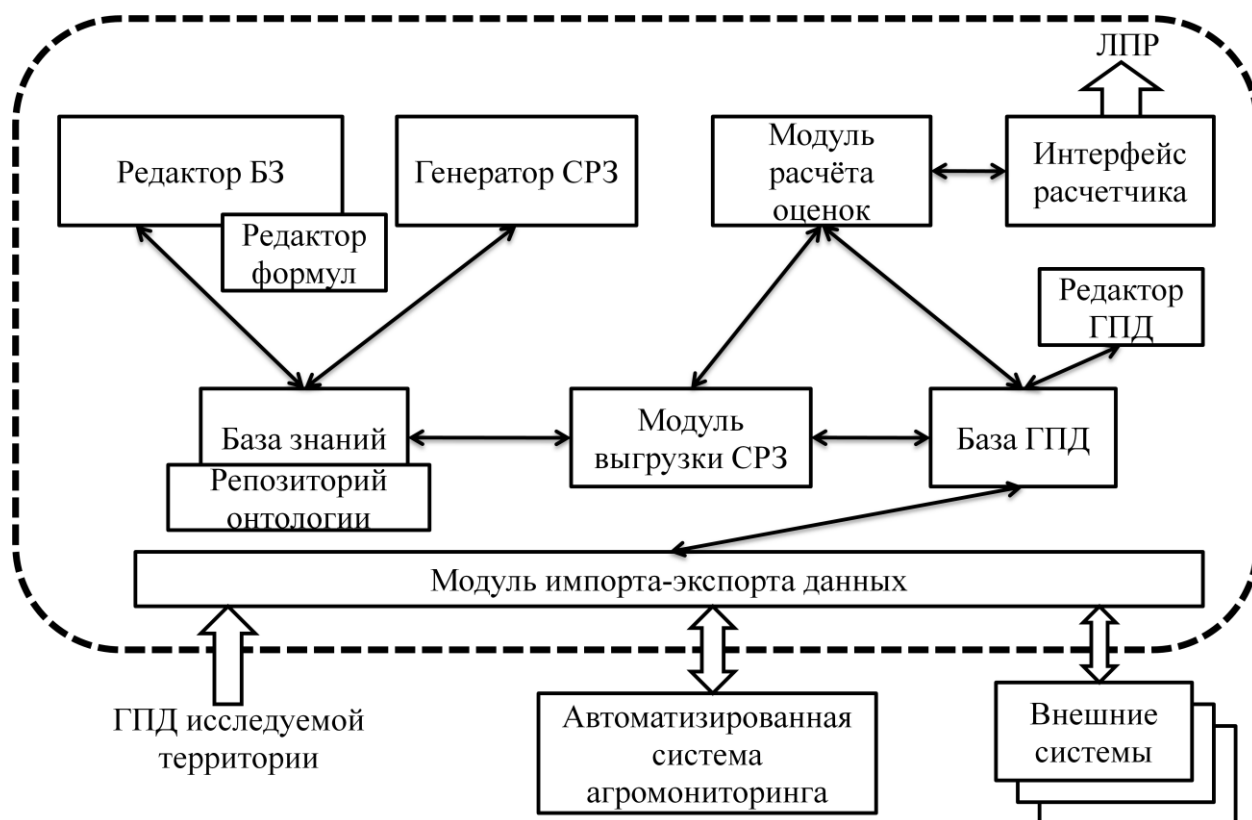


Рисунок 1 — Функциональная структура СППР земель сельскохозяйственного назначения

На рисунке 1 СППР состоит из следующих подсистем:

1) Оценка ЗСХН – это сложный многофакторный анализ, поскольку необходимо учитывать климатические факторы, характеристики почвенного и растительного покровов, особенности инфраструктуры, геопространственные характеристики оцениваемых участков земной поверхности и др. Взвешенные и достоверные оценки базируются на знании экспертов в различных областях, таких как: геоботаника, геодезия, экономика, сельское хозяйство.

2) Построение системы оценок специфично для конкретной решаемой задачи и для конкретной территории, что затрудняет непосредственное привлечение экспертов. Для отчуждения экспертных знаний от их носителей и представления их в формализованном виде, доступном для использования, применяются методы «инженерии знаний». Поэтому база знаний (БЗ) является ключевым блоком системы. Репозиторий онтологий (знания хранятся в системе в форме онтологий), который содержится в БЗ, поддерживает средства обеспечения целостности информации, в том числе средства управления версиями и изменениями.

3) Редактор БЗ позволяет эксперту, ответственному за содержание онтологии, осуществлять диалоговые операции по созданию новой онтологии (версии онтологии) и ее редактированию в части описания таксономии, метрик измерения натуральных параметров объектов оценки и вычислительных соотношений, а также созданию прецедентов.

4) Генератор схемы решения задачи (СРЗ) позволяет эксперту, ответственному за представление задачи, создавать и поддерживать модель описания задачи в рамках определенной версии онтологии, а также осуществлять операции по миграции описания задачи в обновленные версии онтологии.

5) При создании и модификации онтологии возникает процедура описания вычислительных соотношений для метрик оценивания. Данная процедура базируется на использовании редактора формул, который входит как компонента в редактор базы знаний и генератор схемы решения задачи. Помимо базовых математических операций генератор формул содержит возможность формирования таблично заданных функций.

6) Редактор геопространственных данных (ГПД) служит для подготовки и оперирования геопространственным описанием экземпляра решаемой задачи.

7) При создании пустого экземпляра решаемой задачи (модуль 5) модуль выгрузки СРЗ порождает шаблон векторного слоя с предопределенным набором атрибутов, заданных в схеме решения задачи, который помещается в базу ГПД.

8) База ГПД представляет собой набор слоев, наполненных данными о пространственных координатах объектов оценки, тем самым формируется векторное представление данных на анализируемую территорию, содержащее множество объектов оценки как геопространственных объектов.

9) Атрибутивная информация содержит значения натуральных параметров объектов оценки. Поставка атрибутивной информации – ответственность модуля импорта-экспорта данных. Данный модуль взаимодействует с автоматизированной системой агромониторинга и другими внешними системами, что позволяет получать данные на основе обработки и анализа космоснимков, наземных измерений, данных метеостанций и других источников.

10) Расчет промежуточных и финишных оценок осуществляется модулем расчета оценок.

11) Визуализация оценок осуществляется через интерфейс расчетчика, позволяющий формировать пространственное представление данных в виде векторных слоев с вычисленными атрибутами, а также отображать данные в табличной форме либо средствами деловой графики.

В приведенной модели за основу всей системы отвечает модуль БЗ. Именно на основании этого модуля и будет происходить помощь в принятие решения для земель сельскохозяйственного назначения. Подобная система создается на базе лаборатории сибирского федерального университета, кафедры систем искусственного интеллекта и планируется взаимодействие с имеющейся системой агромониторинга.

Структурная СППР в земледелии кандидата технических наук Агрофизического НИИ В.В. Якушева [3] представлена на рисунке 2:

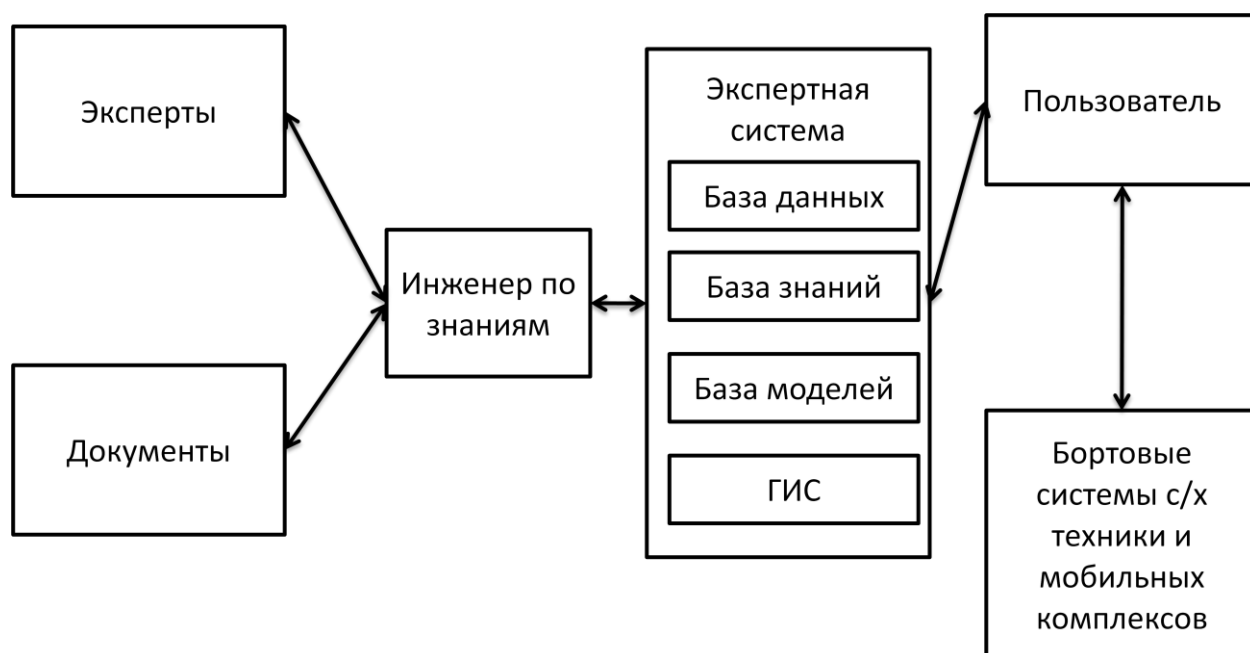


Рисунок 2 — Структура СППР. Автор В.В. Якушев

В своих работах профессор выделяет что эксперт и инженер по знаниям совершенно разные люди. На плечи эксперта ложиться задача по подбору и сопровождению достоверности знаний, в то время как инженер должен лишь только привести знания к одному, единому выводу. В самой же экспертной системе БЗ должна выполнять строго все правила предписанные её, а так же анализировать их.

Эксперты и инженеры по знаниям в данной модели являются ключевыми фигурами, определяющими наполняемость СППР знаниями и данными. Учитывая огромный объем разнородной информации, так или иначе относящейся к сельскому хозяйству, работу по наполнению СППР данными и знаниями можно назвать поистине титанической.

Структурная схема генерации и реализации агротехнологических решений в земледелии представлена на рисунке 3:

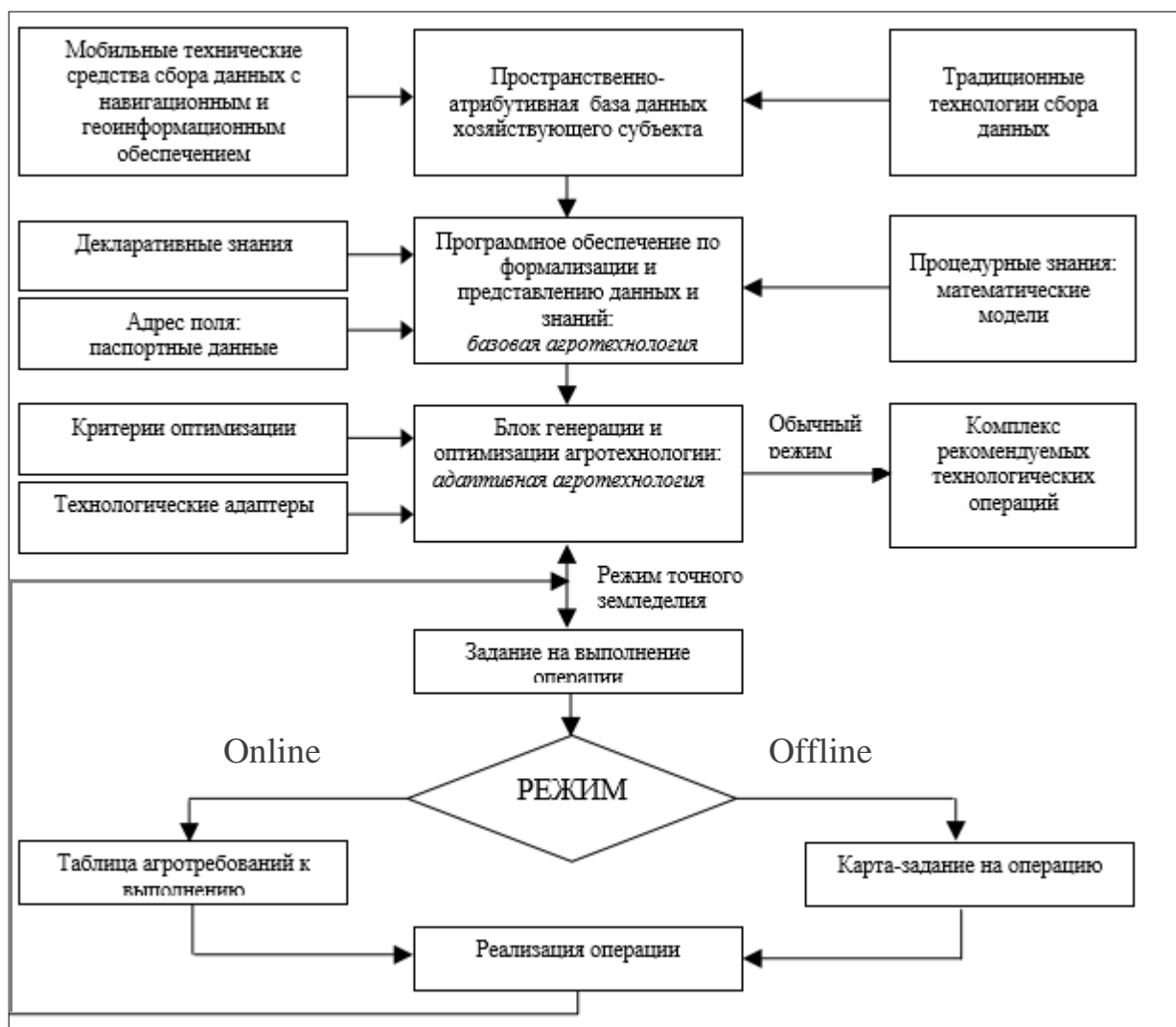


Рисунок 3 — Структурная схема генерации и реализации агротехнологических решений в земледелии

Кандидат технических наук В. И. Вьюн, а так же и её коллеги Г. Е. Кузьменко, и Ю. А. Михненко в статье проблемно-ориентированные базы знаний как платформа саморазвития СППР [4] предлагается устройство базы знаний представленная на рисунке 4.

Таким образом, проблемно-ориентированная база знаний ПО предоставляет возможность отслеживать состояние ПО посредством мониторинга значений факторов, идентифицирующих состояние ПО. При выявлении отклонений значений этих факторов от их нормальных значений можно, автоматически двигаясь по дереву, найти тот сценарий, использование которого может обеспечить предотвращение проблемной ситуации или же ее ликвидацию.

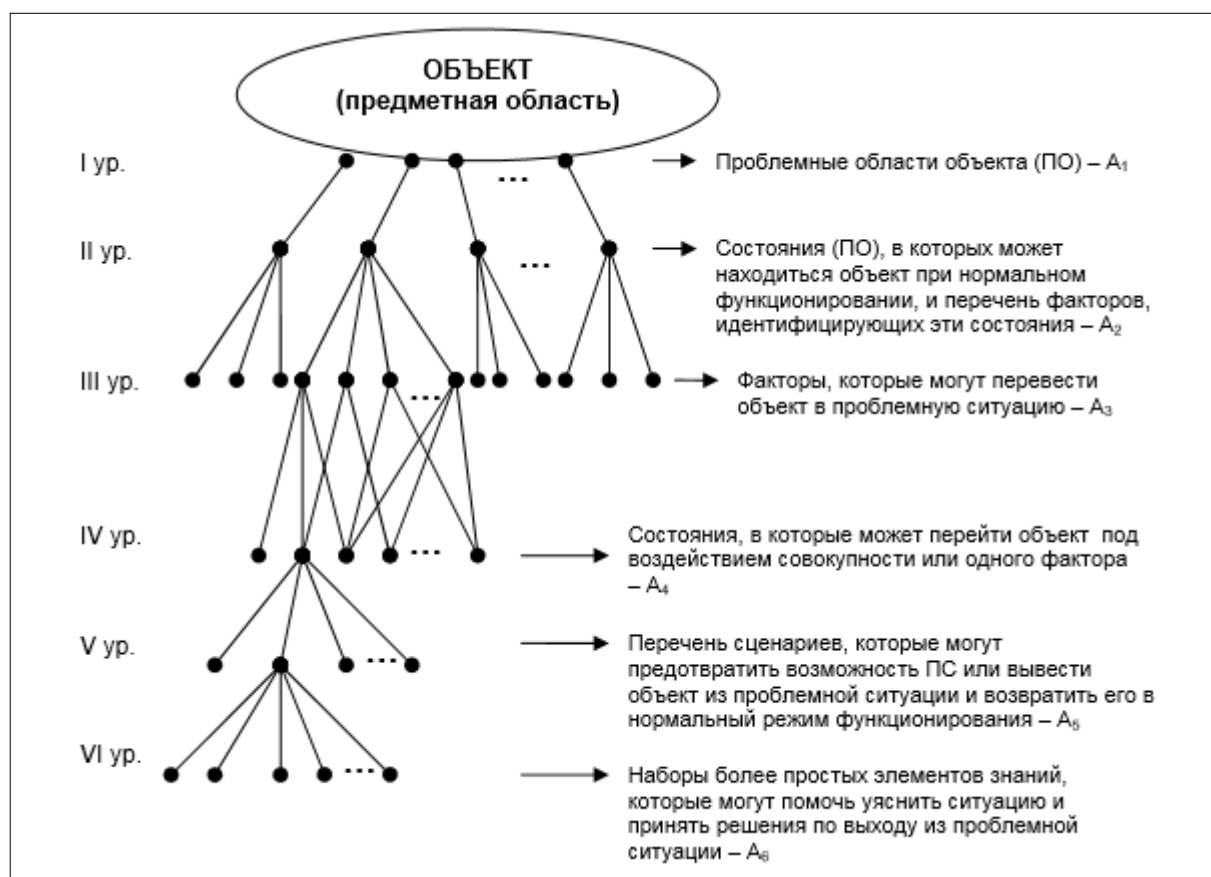


Рисунок 4 — Иерархическая структура проблемно-ориентированной базы знаний

Наряду с традиционными подходами к использованию баз знаний для поддержки непрерывного эволюционного развития системы возникают и неординарные задачи динамического развития системы на основании имеющегося опыта ее функционирования.

1.6 Дескрипционные логики

Семейство языков представления знаний, позволяющих описывать понятия предметной области в недвусмысленном, формализованном виде. Они сочетают в себе, с одной стороны, богатые выразительные возможности, а с другой — хорошие вычислительные свойства, такие как разрешимость и относительно невысокая вычислительная сложность основных логических проблем, что делает возможным их применение на практике. Таким образом, дескрипционные логики (далее ДЛ) представляют собой компромисс между выразительностью и разрешимостью. ДЛ можно рассматривать как разрешимые фрагменты логики предикатов, синтаксически же они близки к модальным логикам.

Свое современное название ДЛ получили в 1980-х. Прежние названия (в хронологическом порядке):

- 1) терминологические системы;
- 2) логики концептов.

Изначально ДЛ зародились как расширение фреймовых структур семантических сетей механизмами формальной логики. В настоящее время ДЛ являются важным понятием в концепции семантической паутины, где их предполагается использовать при построении онтологий. Фрагменты OWL-DL и OWL-Lite языка веб-онтологий OWL также основаны на ДЛ.

Дескрипционные логики оперируют понятиями концепт и роль, соответствующими в других разделах математической логики понятиям «одноместный предикат» (или множество, класс) и «двуместный предикат» (или бинарное отношение). Интуитивно, концепты используются для описания классов некоторых объектов, например, «Люди», «Женщины», «Машины». Роли используются для описания двуместных отношений между объектами, например, на множестве людей имеется двуместное отношение «X есть родитель для Y».

Между людьми и машинами имеется двуместное отношение «X имеет в собственности Y», где в качестве X и Y можно подставить произвольные предметы.

С помощью языка ДЛ можно формулировать утверждения общего вида — о классах вообще (всякая Машина имеется в собственности не более чем у одного Человека) и частного вида — о конкретных объектах (Мария есть Женщина, Иван имеет в собственности Машину 1).

Многочисленные онтологии построены и строятся в самых различных предметных областях, таких как биоинформатика, генетика, медицина, химия, биология. Как только онтология построена, встает вопрос о том, как можно извлекать знания, следующие из содержащихся в онтологии знаний, можно ли это делать программно и каковы соответствующие алгоритмы. Все эти вопросы решаются теоретически в науке «дескрипционная логика», а практически уже реализовано множество программных систем — механизмов рассуждений, которые и позволяют автоматизировано выводить знания из онтологий и производить другие операции с онтологиями.

1.7 Рассуждение по прецедентам

Case-based reasoning, CBR – это способ формирования умозаключений на основании знаний [5]. Он опирается на поиск и анализ ситуаций, когда в прошлом были сформированы подобные умозаключения.

Рассуждения по прецедентам – это способ получения решения при помощи поиска схожих ситуаций, которые хранятся в памяти с прошлым опытом решения задач, и адаптацией найденных решений к изменившимся условиям.

Прецедент в данном случае — это случай или событие, имевшее место в прошлом и служащее примером или основанием для аналогичных действий в настоящем.

Применение CBR для решения задач применяется, когда:

- 1) подобные задачи должны иметь подобные решения (принцип регулярности);
- 2) виды задач, с которыми сталкивается решатель, должны иметь тенденцию к повторению.

CBR в общем случае представляет собой циклический процесс: решение проблемы, запоминание этого решения в качестве прецедента, решение новой проблемы и так далее. CBR-цикл (рисунок 4) может быть описан представленными ниже процессами:

- 1) поиск похожего прецедента;
- 2) адаптация. Осуществляется решение новой задачи на базе имеющегося в системе прецедента. Во время этого процесса совершается проверка полученного решения на корректность и обработку ошибок. Возможна некая коррекция решения.
- 3) Сохранение прецедента, то есть той части полученного опыта, которая может оказаться полезной для решения новых задач (пополнение или корректировка библиотеки прецедентов).

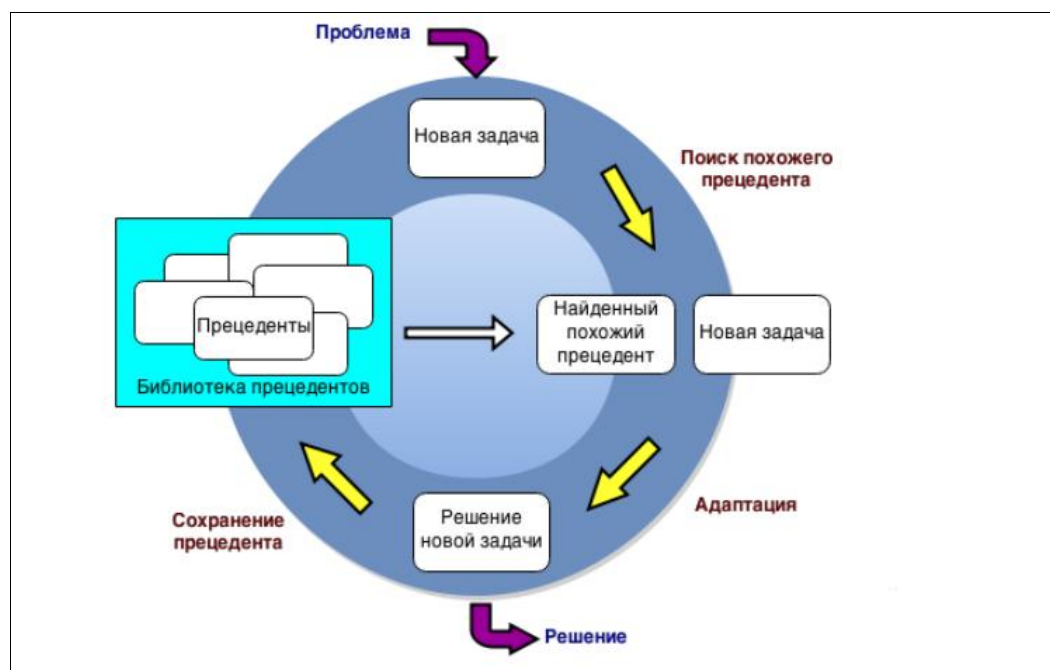


Рисунок 4 — Цикл решения по прецедентам

В данной работе рассуждение по прецедентам использовалось для планирования, то есть для предсказания того, какую культуру сажать при использовании определенных факторов.

Для классификации структуры может различаться два типа. Структура белого ящика повторно используется основным подклассом, а структура черного ящика повторно используется посредством параметризации. Обычная разработка фреймворка начинается с дизайна в виде архитектуры белого ящика, которая развивается в черный ящик. Результирующая структура черного ящика имеет связанный с ней построитель, который будет генерировать код приложения. Визуальный конструктор позволяет разработчику программного обеспечения подключать объекты инфраструктуры и активировать их. Основная идея состоит в разделении основных классов и пользовательского интерфейса. Это разделение даст нам архитектуру двух уровней, показанную на рисунке 5.

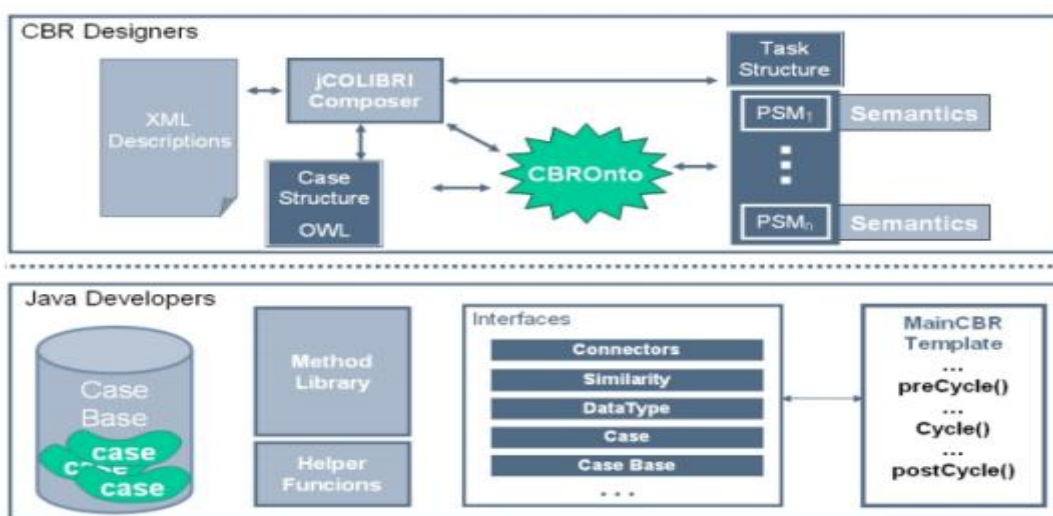


Рисунок 5 — Архитектуры уровней JColibri

Для настройки и вычисления степени сходства заданного прецедента и новой проблемы используем мера подобия (Equal). Простейшая мера подобия – это, по сути, количество общих параметров. Более гибкие меры подобия вычисляют взвешенную сумму общих параметров.

Выводы по главе 1

В данной главе были поставлены цели выпускной квалификационной работы и сформулированы задачи.

Результатом первой задачи: обзора предметной области, в которой рассмотрены СППР основанные на БЗ и было принято решение о том, что дальнейшая работа будет основываться на прецедентах. БЗ СППР созданная на прецедентах для земель сельскохозяйственного назначения необходима для внедрения возможности сравнения текущего состояния земли и её состояния в прошлом, чтобы получить наиболее подходящие параметры её использования с целью извлечь максимальную прибыль при меньших затратах.

В данной главе представлены работы кандидата технических наук Агрофизического НИИ В. В. Якушева, а так же кандидата технических наук В. И. Вьюн и следующих работ [], [], [].

Для фундамента БЗ будет использоваться онтология, созданная в НУЛ «ИПКМ» в системе информационной поддержки космического мониторинга, кафедры систем искусственного интеллекта. Планируется создание прецедентов в онтологии и реализация созданных прецедентов в программном средстве.

Глава 2. Использование программных средств

2.1 Создание прецедентов

Выбранным методом построения базы знаний является рассуждение по прецедентам. В соответствии с данным подходом для того, чтобы разработать базу знаний, необходимо сначала создать прецеденты. Для разработки прецедентов был использован редактор Protege [] версии 5. 2.

Прецедентами являлись даты, полученные из статистики по Сухобузимскому району

Прецеденты служат для документирования функциональных требований к программным системам. Прецедент описывает некоторый целостный фрагмент поведения системы, не вдаваясь при этом в особенности внутренней структуры субъекта. Определение прецедента содержит все свойственные ему виды поведения: основную последовательность, различные варианты стандартного поведения и различные исключительные ситуации с указанием ответной реакции на них. С точки зрения пользователя некоторые из видов поведения выглядят как ошибочные. Однако для системы ошибочная ситуация является одним из вариантов поведения, который должен быть описан и обработан.

При проектировании программной системы производится поиск таких классов для реализации прецедента, которые удачно сочетали бы в себе требуемые роли и не приводили к излишнему усложнению системы. Реализацию прецедента можно смоделировать в виде одной или нескольких коопераций (реализаций прецедента). Один и тот же прецедент может быть описан с различной степенью детализации.

Дальнейшее использование онтологии предполагает применение готового инструментария JColibri, редактора БЗ написанном на языке программирования Java [5]. Онтологическая форма предполагает, что

структура прецедента отображается в онтологию, причем разработка онтологии осуществляется по спецификации OWL2.0 [6], RDF(S).

В состав приложения входит набор пакетов реализующие этапы рассуждения по прецедентам и библиотеку OntoBridge [7]. JColibri является платформой для построения решения, основываясь на онтологиях с дескрипционной логикой.

Для создания прецедентов в Protégé необходимо перейти во вкладку Individuals by class, рисунок 6.

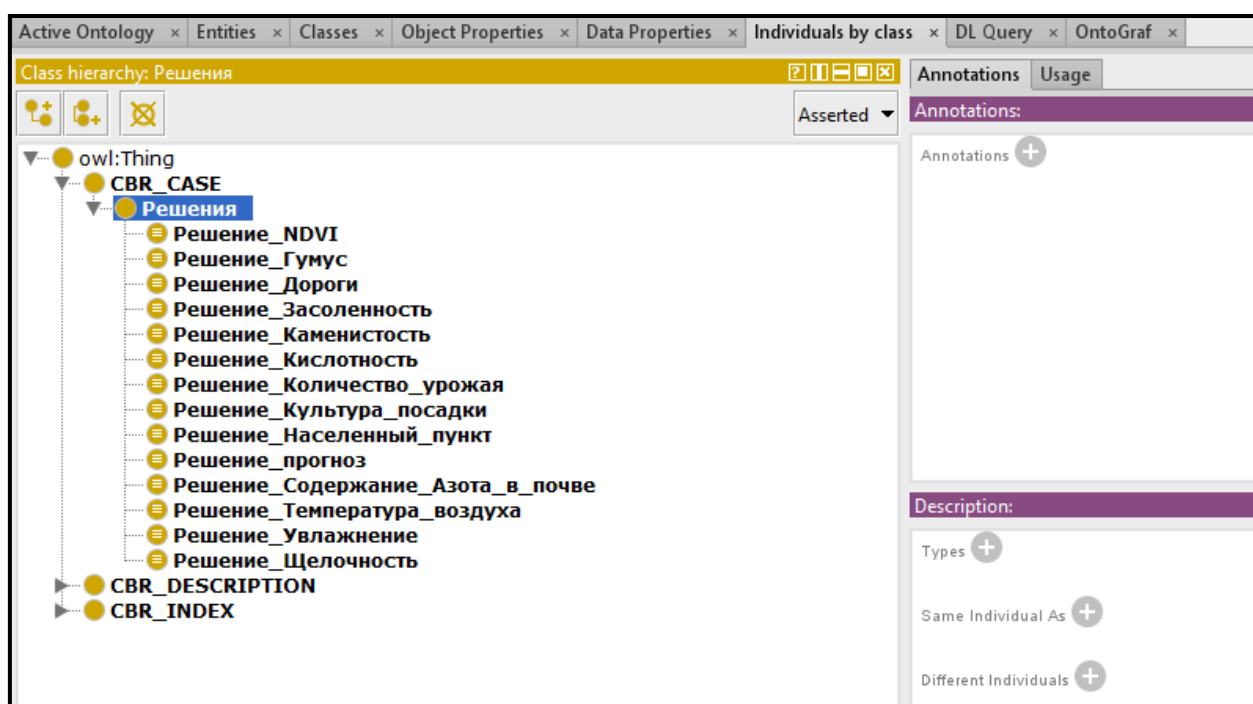


Рисунок 6 — Вкладка Individuals by class

Здесь создается прецедент для класса «Решения» в котором будет заключаться события из прошлого, исходя из которого и будет формироваться решение.

Для создания решения добавим экземпляры прецедентов по годам (рисунок 7).

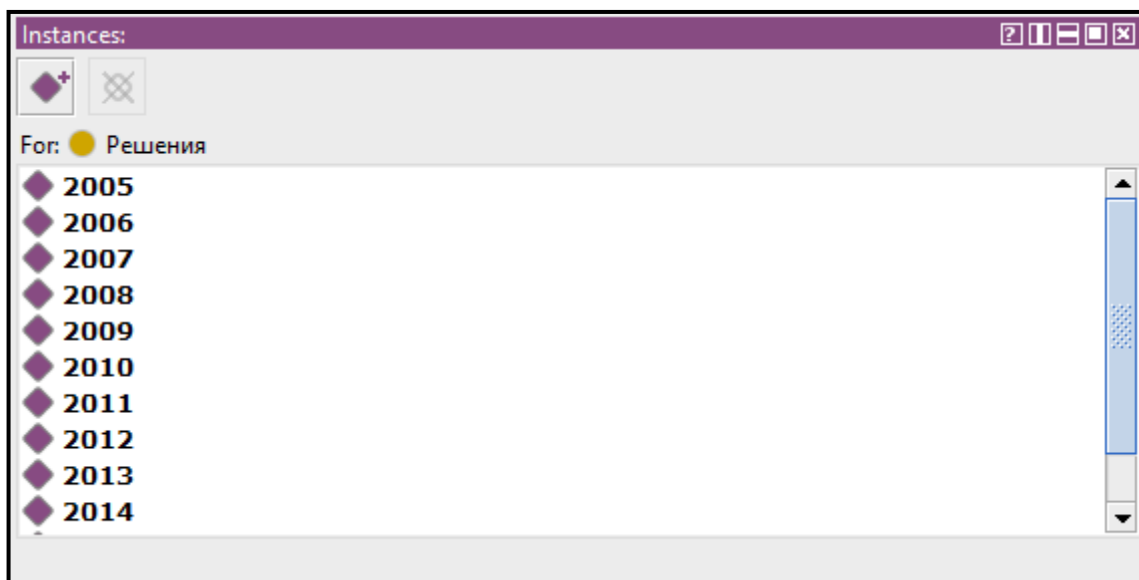


Рисунок 7 — Экземпляры решений

Далее необходимо связать прецеденты по свойствам (рисунок 8). Для этого выбираются все созданные свойства объекта и связываются по их критериям из базы данных.



Рисунок 8 — Свойства объектов прецедентов

В дальнейшем при повторении вышеупомянутых действий формируются правила для создания БЗ. Эти правила будут применяться для принятия решений в программе JColibri.

2.2 Связь свойств прецедентов

Для создания связи у свойств в Protege необходимо обратиться к инструменту записи свойств (рисунок 9).

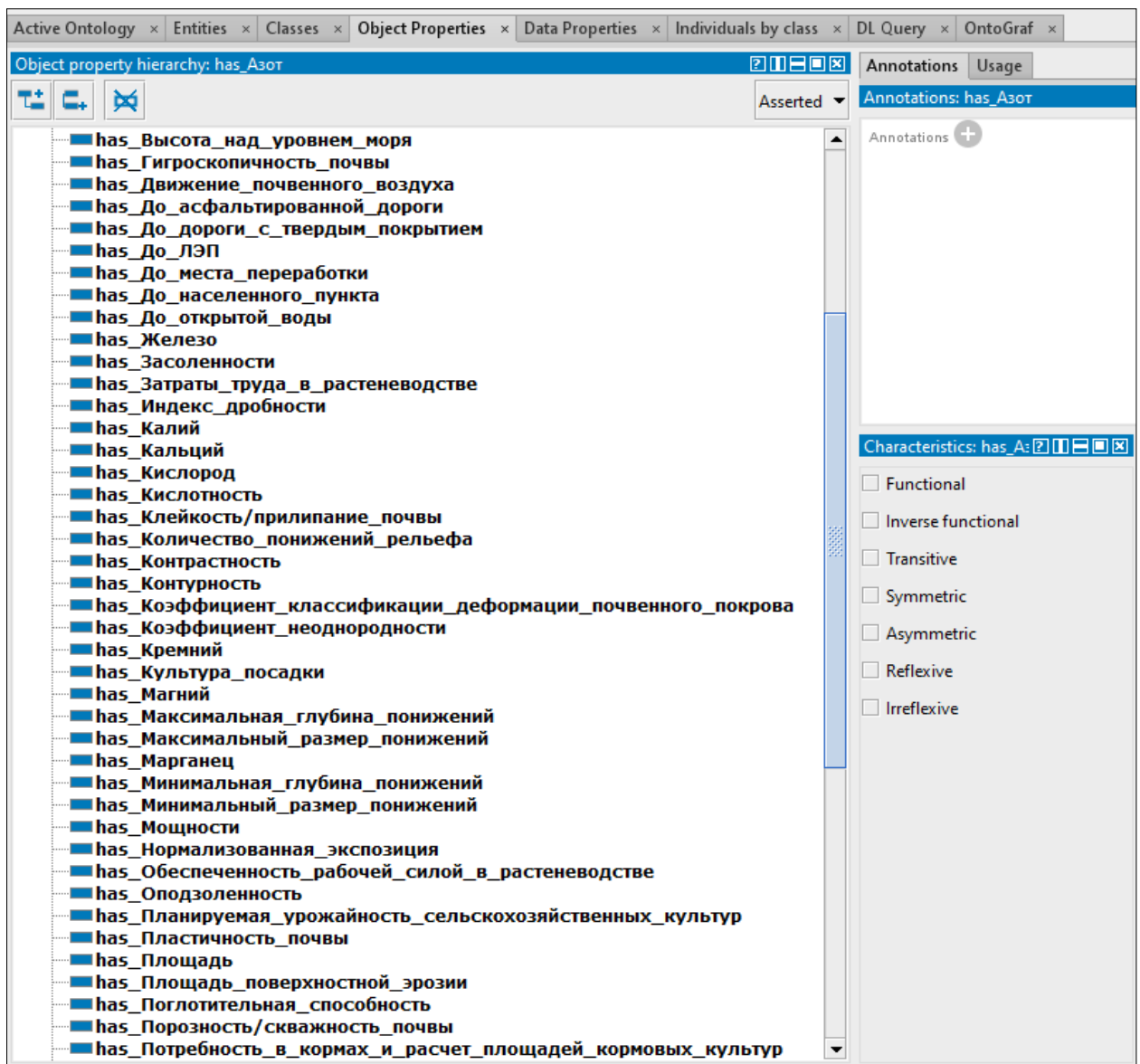


Рисунок 9 — Список доступных свойств

Добавляя свойства для прецедентов необходимо указывать их объектные свойства. В поле Characteristic выбираем по какому отношению будут связаны экземпляры в БЗ.

2.3 Создание системы решения по прецедентам

После создания нового CBR-приложения с расширением Description Logic Extension (использование параметра дескрипционной логики изображено на рисунке 10) происходит настройка структуры онтологии и каждого параметра прецедента.

На основе инструментария и функционала jColibri был сгенерирован класс на языке Java. После подключения всех необходимых библиотек jColibri, было получено успешно функционирующее Java-приложение исходя из работы которого и будет выбран наиболее подходящий прецедент.

При старте программы будет предложен выбор метода анализа БЗ:

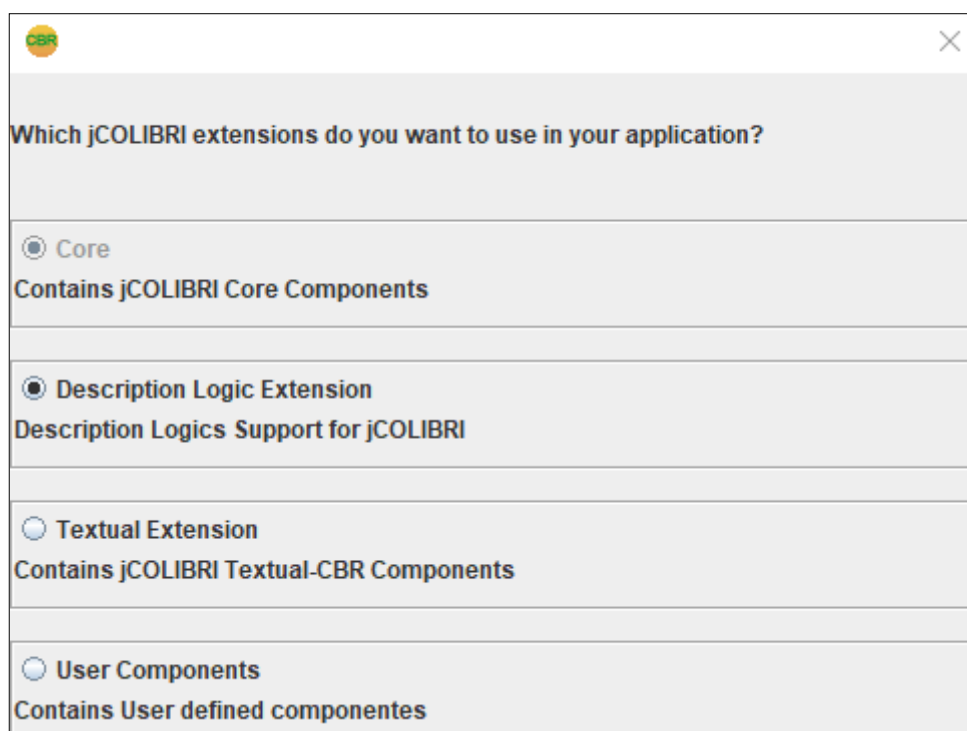


Рисунок 10 — Выбор дескрипционной логики

Для начала создадим структуру принятия решений по созданной ранее онтологии. В поле TYPE выбирается значение ONTOLOGY в соответствии с рисунком 11. После выбора структуры для принятия решений можно будет загрузить ранее созданную или создать её сначала.

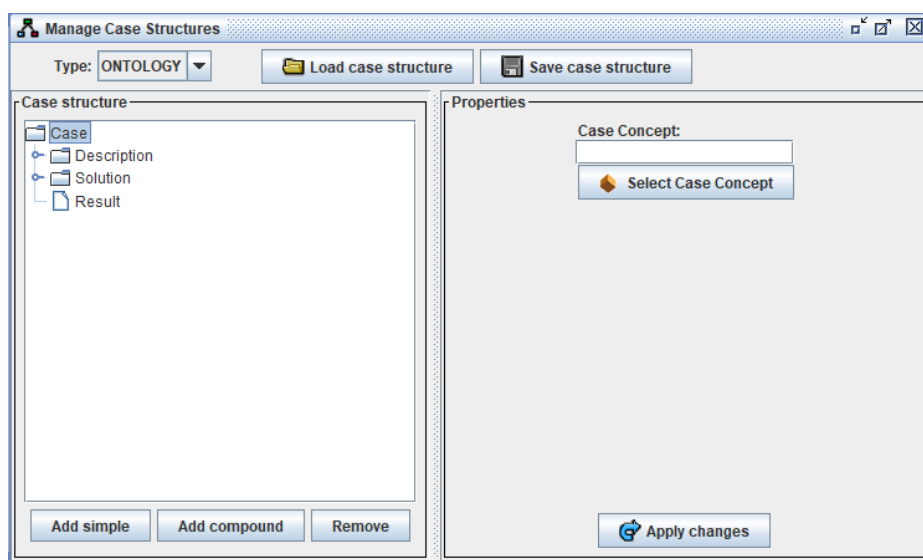


Рисунок 11 — Менеджер структуры принятия решений

После выбирается ранее созданная онтология и считывается. В этот момент все прецеденты из БЗ на основе которых будет сделан выбор переносятся в JColibri, как на рисунке 12.

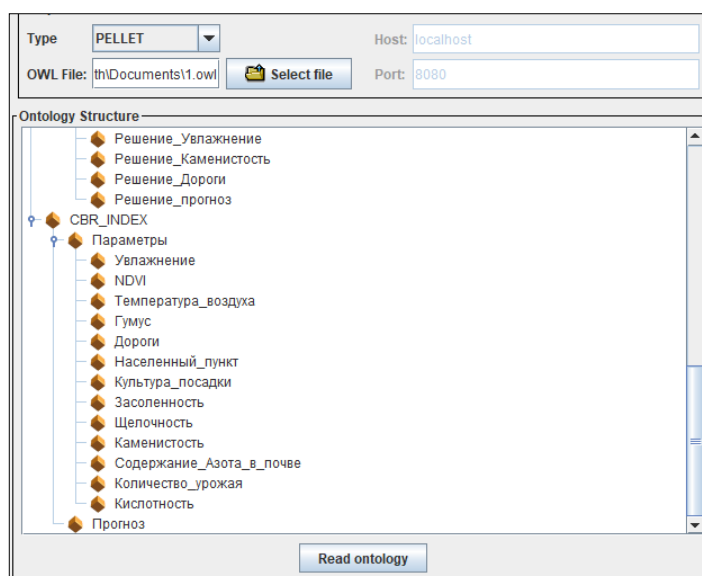


Рисунок 12 — Загрузка структуры онтологии

В дальнейшем для создания решений необходимо заполнить атрибуты (рисунок 13) по которым будет происходить поиск. Значения берутся из онтологии.

Properties

Select Concept

Concept:

Relation:

Weight:

Local similarity:

Similarity parameters

Name	Value
------	-------

Apply changes

Рисунок 13 — Заполнение атрибутов в JColibri

Ставится значение Equal, выбирающее меру близости конкретного атрибута. Это сделано для принятия наиболее подходящего результата при поиске решений. Эту же процедуру повторяем и для других атрибутов. После сохраняется структура БЗ в формате *.xml.

Теперь в древовидной структуре CBR System находим и нажимаем на ветвь иерархии PreCycle. Справа появляются свойства (рисунок 14) выбираем кнопку формирования Instance и у CBR System появляется ещё одна ветвь иерархии Obtain cases task. В её свойствах выделяем метод исследования по ДЛ и нажимаем на кнопку формирования Instance, в запросе файла указываем сохраненный ранее структура решения.

В разделе CBR Cycle также жмём кнопку Instance. Появляется некоторое число вложенных веток.

В ветке CBR Cycle Obtain query task нужно вновь выбрать метод исследования ДЛ и повторить действия, сделанные ранее, нажать Instance и выбрать файл структуры решения.

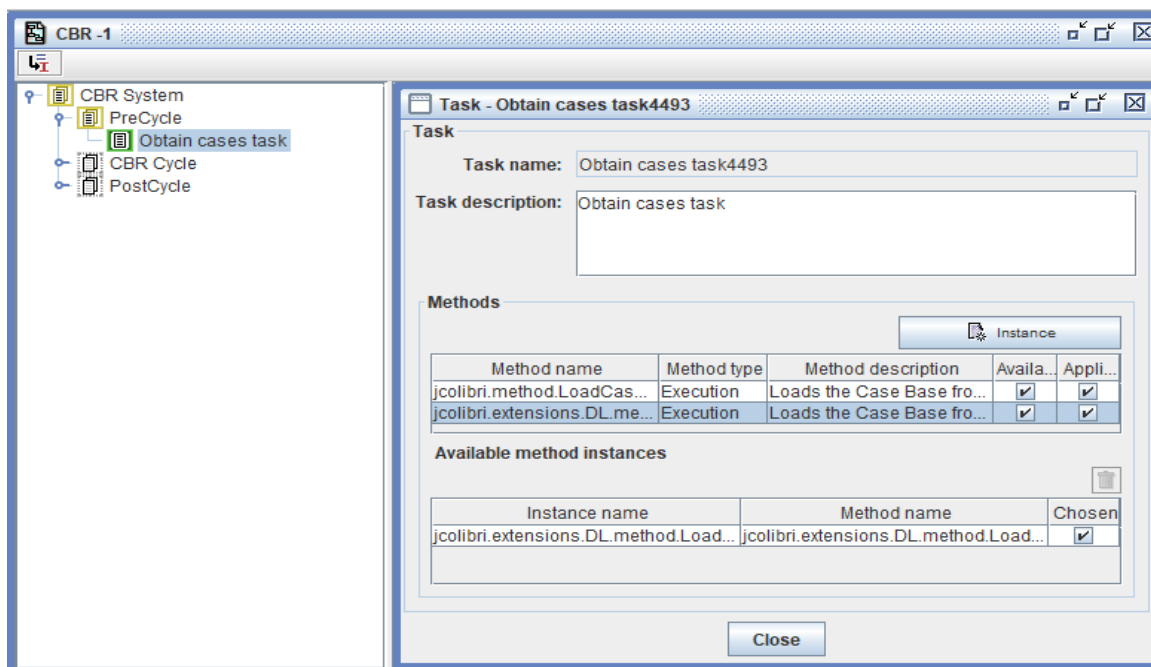


Рисунок 14 — Заполнение дерева принятия решений

В ветке CBR Cycle Retrieve Task также нажать Instance для метода СВЯЗИ КОМПОНЕНТОВ ОНТОЛОГИИ.

У ветки CBR Cycle Retrieve Task Select working cases task нажать Instance для выбора места хранения прецедентов.

Для следующего раздела необходимо выбрать CBR Cycle Retrieve Task Compute Similarity Task также нажать Instance для метода построения структуры решений.

При выборе следующего метода мы выбираем CBR Cycle Retrieve Task Select best task и создаем с помощью кнопки Instance способ выбора знаний по наилучшему совпадению.

Таким образом настроив CBR систему на чтение онтологии посредством дескрипционной логики мы можем приступить к её запуску.

В итоге получается следующее дерево, изображенное на рисунке 15:

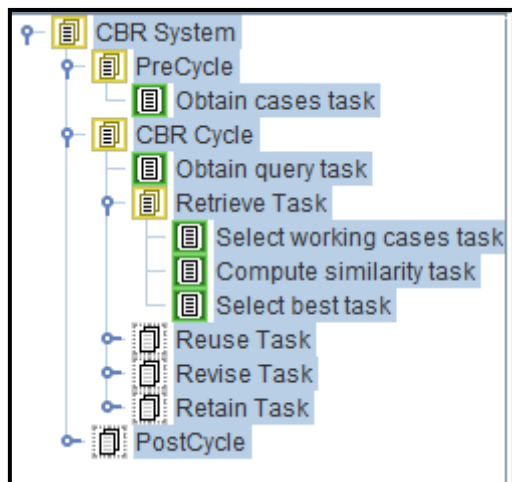


Рисунок 15 — Результат заполнения дерева

2.4 Результат работы программы

Для принятия решений необходимо выделить всё дерево (рисунок 15) и нажать кнопку Solve to для компоновки всех введенных значений и построения решения на основании онтологии. С помощью ДЛ создается структура прецедентов для вывода параметров по принимаемому решению.

После вычисления совпадений по событиям из прошлого выводятся строки со значением максимально приближенным к единице, т. к. единица это показатель максимального совпадения прецедента с заданными параметрами. Ранее мы выбирали метод наилучшего совпадения, в связи с этим нам выводятся результаты в диапазоне от нуля до единицы.

Далее необходимо ввести параметры, которые ЛПР желает получить при помощи в принятии решения, это действие изображено на рисунке 16.

Чтобы выбрать необходимый параметр в БЗ необходимо оставить значение «<empty>» и сдвинуть ползунок к значению «0».

Увлажнение	<empty>	1.0
NDVI	<empty>	1.0
Температура_воздуха	<empty>	1.0
Гумус	<empty>	1.0
Дороги	<empty>	1.0
Населенный_пункт	<empty>	1.0
Культура_посадки	<empty>	1.0
Засоленность	<empty>	1.0
Щелочность	<empty>	1.0
Каменистость	<empty>	1.0
Содержание_Азота_в_почве	<empty>	1.0
Количество_урожая	<empty>	1.0
Кислотность	<empty>	1.0
Прогноз	<empty>	1.0

Ok Cancel

Рисунок 16 — Окно выбора параметров для принятия решений

После прохождения по всем прецедентам нам выводится наиболее соответствующий Case и отчет в виде представленным на рисунке 17:

```
[SelectAllCasesMethod] INFO: Selecting all cases.  
[NumericSimComputationMethod] INFO: Similarity with case Среднезасоленная :0.0  
[NumericSimComputationMethod] INFO: Similarity with case :0.35714285714285715  
[NumericSimComputationMethod] INFO: Similarity with case :0.35714285714285715  
[NumericSimComputationMethod] INFO: Similarity with case :0.5714285714285714  
[NumericSimComputationMethod] INFO: Similarity with case :0.2857142857142857  
[NumericSimComputationMethod] INFO: Similarity with case :0.2857142857142857  
[NumericSimComputationMethod] INFO: Similarity with case :0.42857142857142855  
[NumericSimComputationMethod] INFO: Similarity with case :0.5714285714285714  
[NumericSimComputationMethod] INFO: Similarity with case :0.5714285714285714  
[NumericSimComputationMethod] INFO: Similarity with case :0.2857142857142857  
[NumericSimComputationMethod] INFO: Similarity with case :0.21428571428571427  
[NumericSimComputationMethod] INFO: Similarity with case :0.2857142857142857  
[NumericSimComputationMethod] INFO: Similarity with case :0.21428571428571427
```

Рисунок 17 — Результат просмотра прецедентов

Также выводится наиболее подходящий результат соответствующий нашим запросам изображенный на рисунке 18.

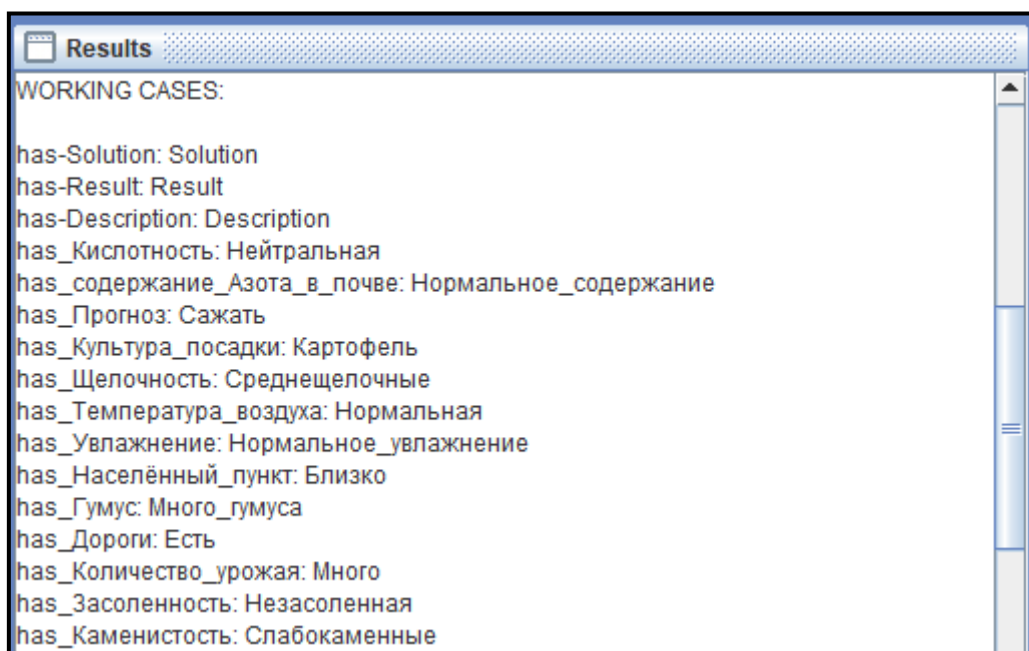


Рисунок 18 — Вывод результата

JColibri определила, что прецедентом с наиболее близкими параметрами является кейс со значением близости 0.57, что и было показано после надписи WORKING CASES.

А это значит, что исходя из предыдущего опыта, по текущим запросам был поставлен прогноз «Сажать» будет «Много урожая».

Выводы по главе 2

Исходя из онтологии, созданной студенткой Мощенковой Д. С. ,были созданы прецеденты и заданы свойства для них. Прецеденты создавались так же опираясь на имеющуюся базу данных по Сухобузимскому району, и являются достоверными.

Создана система рассуждения по прецедентам помогающая принять решение на основе онтологической БЗ. Создание выполнено с помощью программного средства JColibri.

Все созданные прецеденты были реализованы. Результаты работы программы представляются в виде выбора наиболее подходящего прецедента из БЗ. Данные полученные в ходе работы рассуждений по прецедентам в дальнейшем можно использовать для СППР в сельскохозяйственном производстве.

ЗАКЛЮЧЕНИЕ

В рамках бакалаврской работы цель, поставленная на ее выполнение была достигнута. Создана база знаний для систем поддержки принятия решений в сельскохозяйственном производстве.

Выполнены задачи поставленные для достижения цели работы: выполнен обзор предметной области, созданы прецеденты на основе данных из прошлого и произведена их реализация посредством программного обеспечения в базу знаний. В дальнейшем полученные результаты планируется использовать в создании полноценной СППР в рамках лаборатории кафедры систем искусственного интеллекта.

Статья по теме данной выпускной квалификационной работы опубликована в Международной научно-технической конференции студентов, аспирантов и молодых ученых «Перспективы — 2017», посвященная Году экологии в Российской Федерации.

Онтология ЗСХН для СППР в сельскохозяйственном производстве разработана при выполнении бакалаврской работы по направлению 09.03.02 « Информационные системы и технологии», по профилю подготовки 09.03.02.04 «Информационные системы и технологии в медиаиндустрии»

СПИСОК СОКРАЩЕНИЙ

СППР — Система поддержки принятия решения

GPS — Глобальные систем позиционирования

ГИС — Геоинформационные технологий

БЗ — База знаний

ЗСХН — Земли сельскохозяйственного назначения

СРЗ — Схема решения задач

ГПД — Геопространственные данные

ЭС — Экспертная система

ДЛ — Дескрипционные логики

СВР — Рассуждение про прецедентам

ЛПР — Лицо принимающее решение

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кривоконева, Е. Ю. Мониторинг земель с применением ГИС-технологий / Кривоконева Е. Ю., Гончарова И. Ю. // Научный журнал Российского НИИ проблем мелиорации, № 4(04), 2011 г.
2. Якушев, В. В. Система поддержки принятия решений в земледелии. Принципы построения и функциональные возможности. // Журнал известия российского государственного педагогического университета им. А. И. Герцена. 2009. С. 110-114.
3. Беглов, И. Ф. Система поддержки принятия решений в водно-экологическом секторе бассейна Аральского моря / Беглов И. Ф., Сорокин А. Г. // Научно информационный центр МКВК. 2015.
4. Михненко, Ю. А. Проблемно-ориентированные базы знаний – платформа саморазвития СППР. / Вьюн В. И., Кузьменко Г. Е., Михненко Ю. А. // Журнал Математичні машини і системи, 2009, № 4 С. 52–57
5. JColibri 1.1 Tutorial [Электронный ресурс] : официальный сайт разработчиков. Инструкция по пользованию программой. — Режим доступа: <http://sourceforge.net>.
6. Спецификация OWL 1.1 [Электронный ресурс] — Режим доступа: http://www.w3.org/2006/04/OWL_UseCases-ru.html
7. Библиотека OntoBridge [Электронный ресурс]. официальный сайт разработчиков. Инструкция по пользованию программой. — Режим доступа: <http://sourceforge.net>.
8. Раевич, К. В. Система поддержки принятия решений по оцениванию сельскохозяйственных угодий / Раевич К. В., Маглинец Ю.А. // Современные проблемы дистанционного зондирования Земли из космоса: сб. тезисов XII открытой Всерос. конф. (Москва, 10–14 ноября 2014 г.). М.: Изд-во ИКИ РАН, 2014. С. 394.

9. Раевич, К. В. Модель представления информации о состоянии и динамике земель сельскохозяйственного назначения / Раевич К. В., Маглинец Ю.А., Цибульский, Г.М. // Журнал Сибирского федерального университета. Серия: Техника и технологии. 2014. Т. 7. № 8. С. 984–989.
10. Раевич, К. В. Интеллектуальная система поддержки принятия управленческих решений в задачах оценки земель сельскохозяйственного назначения / Раевич К. В., Зеньков И. В. // Журнал Информатика, вычислительная техника и управление. 2016. № 5. С. 95-104.
11. Ларичев, О. И. Системы поддержки принятия решений. Современное состояние и перспективы их развития : статья / О. И. Ларичев, А. Б. Петровский // Журнал «Итоги науки и техники». Сер. Техническая кибернетика. — Т.21. М.: ВИНТИ, 1987, с. 131—164.
12. Мирсаидов, С.А. Современные концепции экономико-экологического землепользования / Мирсаидов С.А., Нурбоев Н.М. // Электронный журнал Экономическая социология, 2008. Т. 9. № 1. С. 20–50.
13. Баранов, В. Д. Программирование урожаев сельскохозяйственных культур / Баранов В. Д., Тараканов И. Г. // Учеб. пособие — М.: Изд-во УДН, 1990 — 71с.

ПРИЛОЖЕНИЕ А

Плакаты презентации

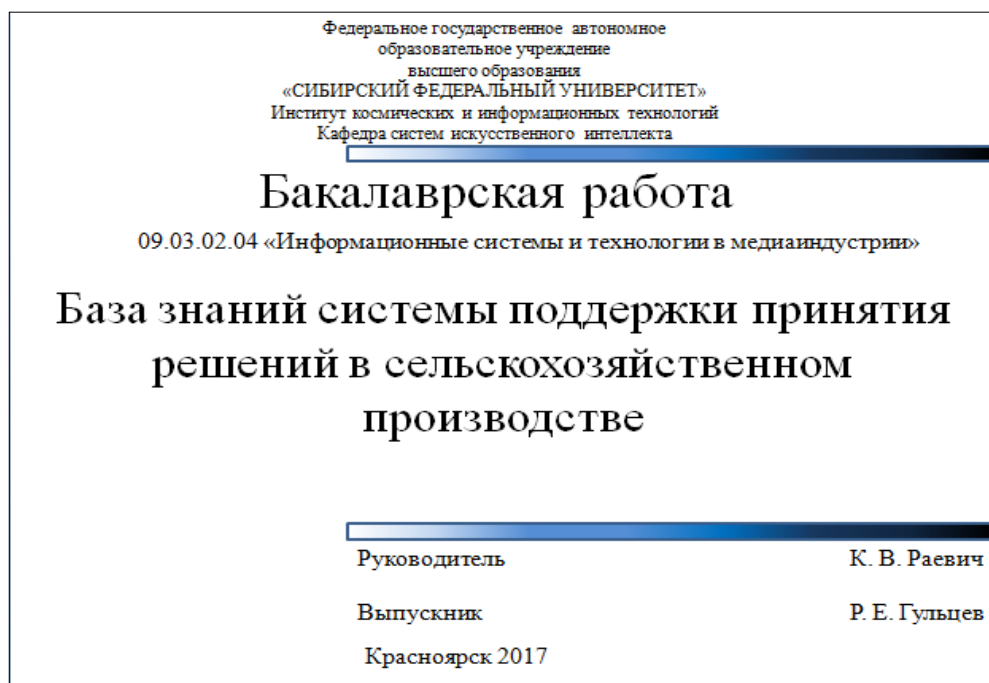


Рисунок А. 1. — Слайд презентации № 1

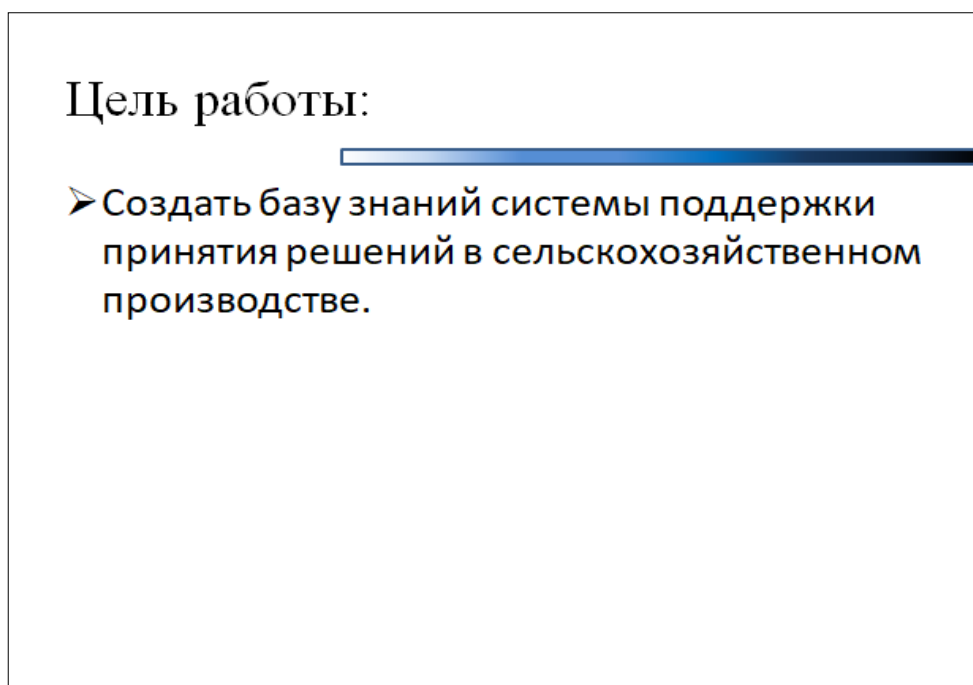


Рисунок А. 2. — Слайд презентации № 2

Задачи:

- ✓ Обзор предметной области;
- ✓ Создание прецедентов для БЗ;
- ✓ Реализация прецедентов БЗ в программном средстве JColibri.

Рисунок А. 3. — Слайд презентации № 3

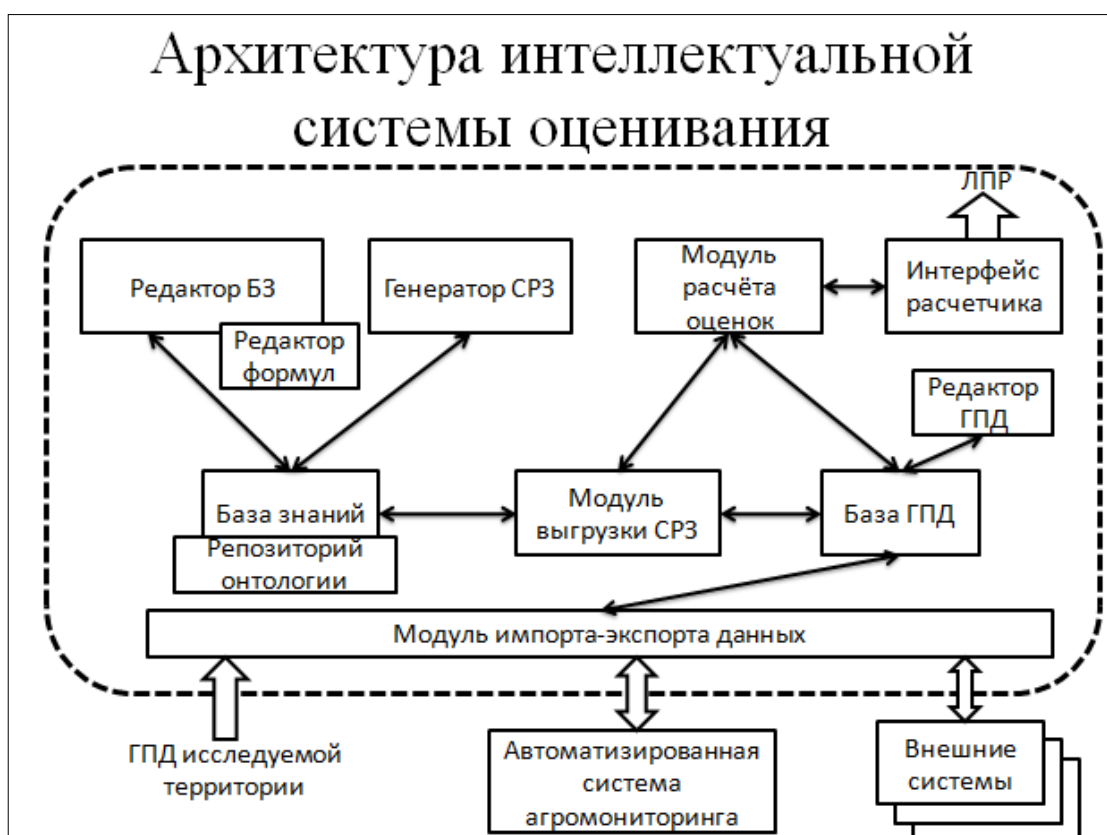


Рисунок А. 4. — Слайд презентации № 4

Рассуждение на основе прецедентов



Рисунок А. 5. — Слайд презентации № 5

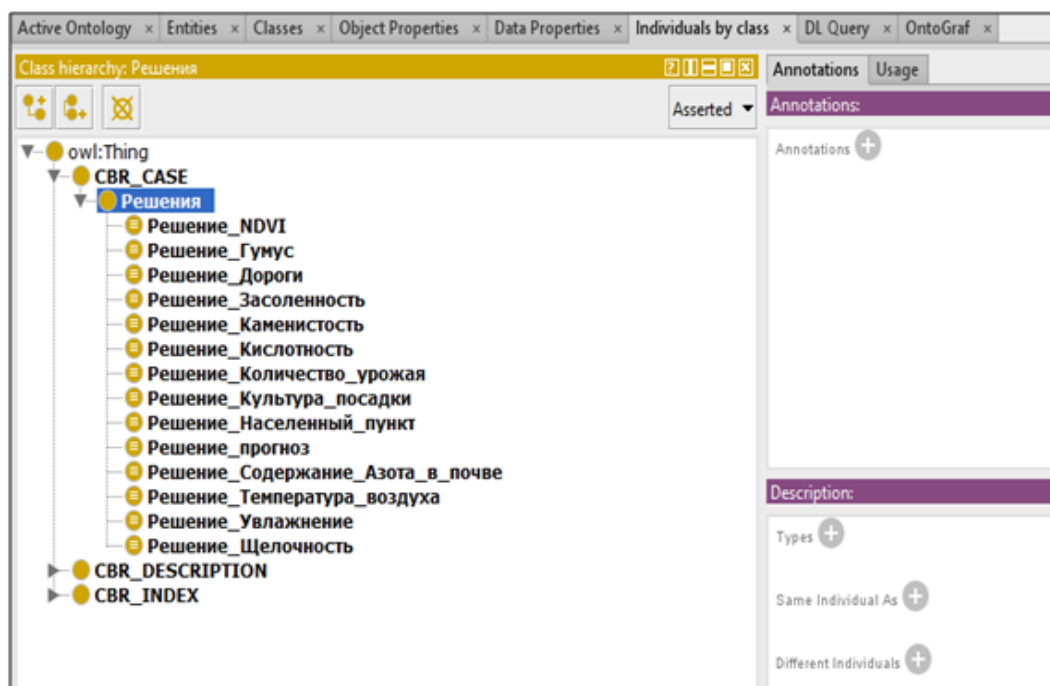


Рисунок А. 6. — Слайд презентации № 6

Свойства экземпляров в базе знаний

Object property assertions +	
has_Прогноз Сажать	? @ x o
has_Гумус Норма_гумуса	? @ x o
has_Населённый_пункт Далеко	? @ x o
has_Культура_посадки Картофель	? @ x o
has_Температура_воздуха Нормальная	? @ x o
has_Количество_урожая Много	? @ x o
has_Увлажнение Среднее_увлажнение	? @ x o
has_Щелочность Слабощелочные	? @ x o
has_NDVI 40-60	? @ x o
has_Дороги Есть	? @ x o
has_содержание_Азота_в_почве Нормальное_содержание	? @ x o

Рисунок А. 7. — Слайд презентации № 7

Форма заполнения параметров

Query

Requested parameters

Увлажнение	<empty>	1.0
NDVI	<empty>	1.0
Температура_воздуха	<empty>	1.0
Гумус	<empty>	1.0
Дороги	<empty>	1.0
Населенный_пункт	<empty>	1.0
Культура_посадки	<empty>	1.0
Засоленность	<empty>	1.0
Щелочность	<empty>	1.0
Каменистость	<empty>	1.0
Содержание_Азота_в_почве	<empty>	1.0
Количество_урожая	<empty>	1.0
Кислотность	<empty>	1.0
Прогноз	<empty>	1.0

Ok Cancel

Рисунок А. 8. — Слайд презентации № 8

Вывод расчетов программы

```
[SelectAllCasesMethod] INFO: Selecting all cases.  
[NumericSimComputationMethod] INFO: Similarity with case Среднезасоленная :0.0  
[NumericSimComputationMethod] INFO: Similarity with case :0.35714285714285715  
[NumericSimComputationMethod] INFO: Similarity with case :0.35714285714285715  
[NumericSimComputationMethod] INFO: Similarity with case :0.5714285714285714  
[NumericSimComputationMethod] INFO: Similarity with case :0.2857142857142857  
[NumericSimComputationMethod] INFO: Similarity with case :0.2857142857142857  
[NumericSimComputationMethod] INFO: Similarity with case :0.42857142857142855  
[NumericSimComputationMethod] INFO: Similarity with case :0.5714285714285714  
[NumericSimComputationMethod] INFO: Similarity with case :0.5714285714285714  
[NumericSimComputationMethod] INFO: Similarity with case :0.2857142857142857  
[NumericSimComputationMethod] INFO: Similarity with case :0.21428571428571427  
[NumericSimComputationMethod] INFO: Similarity with case :0.2857142857142857  
[NumericSimComputationMethod] INFO: Similarity with case :0.21428571428571427
```

Рисунок А. 9. — Слайд презентации № 9

Результат работы программы

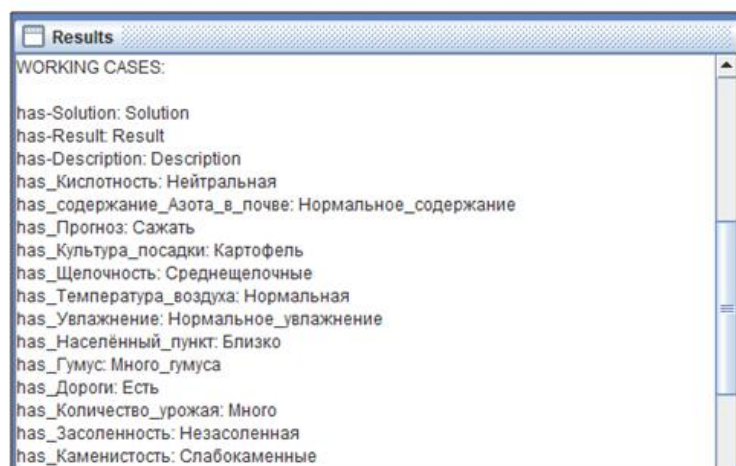


Рисунок А. 10. — Слайд презентации № 10

Выводы:

- ✓ произведен обзор предметной области;
- ✓ создана база знаний на основе прецедентов;
- ✓ прецеденты реализованы в программном средстве с возможностью вывода результата по заданным параметрам;
- ✓ результаты работы были представлены на конференции и обсуждены на конференции «Молодёжь и наука 20/17 века» (17 апреля 2017 г. ФГАОУ ВО СФУ)

Рисунок А. 11. — Слайд презентации № 11

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

Бакалаврская работа

09.03.02.04 «Информационные системы и технологии в медиаиндустрии»

База знаний системы поддержки принятия решений в сельскохозяйственном производстве

Руководитель	К. В. Раевич
Выпускник	Р. Е. Гульцев
Красноярск 2017	

Рисунок А. 12. — Слайд презентации № 12

ПРИЛОЖЕНИЕ Б
Сертификат об участии в конференции

